FASTER — A TOOL FOR DSN FORECASTING AND SCHEDULING

David Werntz, Steven Loyola, and Silvino Zendejas
Jet Propulsion Laboratory
Pasadena, California

Abstract

FASTER (Forecasting And Scheduling Tool for. Earth-based Resources) is a suite of tools designed for forecasting and scheduling of JPL's Deep Space Network (DSN). The DSN is a sc(of antennas and other associated resources that must be scheduled for satellite communicant ions, astronomy, maintenance, and testing. FASTER is a MS-Windows based program that replaces IWO existing programs (RALPH and PC4CAST). FASTER was designed to be more flexible, maintainable, and user friendly. FASTER makes heavy use of commercial software to allow for customization by users. FASTER implements scheduling as a two pass process: the first pass calculates a predictive profile of resource utilization; the second pass uses this information to calculate a cost function used in a dynamic programming optimization step. This information allows the scheduler to "look ahead" at activities that are not as yet scheduled. FASTER has succeeded in allowing wider access to data and tools, reducing the amount of effort expended and increasing the quality of analysis.

Background and History

FASTER, the Forecasting And Scheduling Tool for Iiarlh-based Resources is a suite of software tools developed at JPL (Jet Propulsion Laboratory) to aid in the process of allocating DSN (Deep Space Network) 70 and 34 meter antennas and equipment to track deep space satellites (e.g., Galileo and Voyager) and ground based astronomy (e.g., Goldstone Solar System Radar). In addition, preventive maintenance, testing, and upgrades must be scheduled,

The 70 and 34 meter networks of the DSN presently include a total of 9 antennas (with plans for 3 to 9 more antennas in the next ten years), In addition, there are multiple transmitters, receivers and other pieces of equipment that must be coordinated with the antennas. These nine antennas arc located at three sites around the world: Goldstone, California; Canberra, Australia; and Madrid, Spain (chosen to allow continuous coverage of deep-space objects near the ecliptic). Each site has three different type of antennas — a 70 meter (70 M), a 34 meter standard (34 S), and a 34 meter high efficiency (34H).

Normal activities that utilize the antennas (i.e. not maintenance type activities) generally have three parts:

- *Pre-calibration* — this time is dedicated to configuring the antenna, testing the configuration, and calibrating equipment. This may occur while the object to be tracked is not in view. Typical] y pre-calibration times run between 45 minutes and onc and onc half hours, The time required is determined by"' the type of tracking to be performed and the equipment required.

- *Track* — this is the time during which actual transmission and/or reception occurs, The object to be tracked must be in view during this time. Typical times range from two to ten hours. The amount of time required is determined by data rates, data quantity, desired risk, and many other factors.

- *Post-Calibration* — this time is dedicated to de-configuring the antenna and its associated equipment. Typical times range from 15 minutes to 45 minutes and are determined by the configuration. This may occur while the object is not in view

Under normal circumstances these three parts occur consecutively with no intervening time. Taken together, these three parts comprise a *pass;* however, we will often use the generic term *activity* to mean a pass or other use of an antenna and equipment (e.g., maintenance),

The previous paragraph referred to an object as being *in view,* This implies that an object is "visible" from the antenna. Each time during which an object is in view is known as a view *period.* A view period is said to have a rise (the start of the view period) and a set (the end of the view period). The calculation of view periods also involves information about surrounding terrain, antenna geometry, and signal to noise limitations. These calculations arc performed by a program external to FASTER and stored in a database. For the majority of the deep space objects, this all translates to having one, approximately 10 hour view period per site per day that varies seasonally (in time of day and duration).

Schedules arc segmented into weekly pieces. Initial schedules, with conflicts in them, arc generated six months to two years in advance of real time. This allows time for negotiation of conflicts and for other more detailed planning functions to occur (e.g. detailed spacecraft sequence generat ion). In general, schedules are resolved until they arc conflict frcc, two months in advance of real time.

Currently, approximately 200 to 300 activities arc schcdulcd weekly for thcsc antennas and the numbers continue to grow. On average, only 50-60% of rcquircmcnts are met and any improvcmcnt in scheduling efficiency can result in significant additional science ret urn.

While the goal in scheduling is to gcncratc schcdulcs that utilize the resources as efficiently as possible, the goal in forecasting is to help set thc stage so that scheduling is easier and more efficient. Forecasting can bc thought of as a “what if” capability. Someone poses a change and would like to know the impact of that change. While it is true that one could generate schedules and evaluate schcdulcs, it is not necessarily feasible. Often, the level of detail ncccssary to generate a schedule dots not exist (e.g., in the case of a spacecraft that is just in the design phase). Another problem is that the change should bc evaluated over a fairly long period of time (ycars). Even if schedules could be crcatcd at thc ratc of onc per minutc, a four year study would take over four hours to gcncratc. Finally, it can bc difficult to generalize from a schedule; it is very easy to infer data from a detailed schedule that is really only an artifact of that particular schedule solution,

There are a variety of questions that arc posed as “what-if” studies; thc processing of theses studies has become a major consumer of effort. Some examples include:

- What would bc the impact of cutting prc-calibration time in half?
- If another antenna were to bc added to the network, at which of the three sites should it bc added to do the most good?
- If project A were funded, do wc have enough rcsourccs to support it and all other missions that arc already funded? If not, what might get cut?

In the area of long term forecasting, lead times for both resource augmentation and project (spacecraft) dcvclopmcnt arc measured in years and costs are in 100’s of millions of dollars; therefore, it is of utmost importance to properly answer these questions. Currently, forecasting is performed as far out as 20 years and as close to real lime as two years.

Since 1986, the RALPH (Resource Allocation and Planning Helper) system has been operationally used to assist in the scheduling process. RALPH runs on VAX hardware under VMS and has primarily a character-based intcrfacc for usc on dumb terminals. In 1989, a prototype, called PC4CAST, was developed to assist in forecasting tasks. PC4CAST ran on PC compatible hardware under DOS and used Quattro Pro (a spreadsheet) for tabular data entry and for results graphing. The majority of calculations were done by an external program, reading and writing Quattro data files. RALPH and PC4CAST had no linkagc and because of the different platforms, some data had to be duplicated.

Algorithms

There arc two algorithms of interest in FASTER, generically known as the first and second pass algorithms. The scheduling tool uses both algorithms, while the forecasting tool uses only the first pass. In the first pass, FASTER creates a set of *expected usage profiles,* which represent a statistical analysis of resource demand. The second pass uses this information to derive a cost function which drives a dynamic programming algorithm for group activity scheduling. Group activity scheduling implies that activities are not scheduled individually but rathcr that a set of relatively homogcnous activities arc scheduled as a whole, A group of activities is oftcn referred to as one request or project requirement. A typical requirement might be that Pioneer 10 wants seven, four to eight hours tracks on any 70M antenna and the tracks must bc separated by at least 10 hours and by no more than 36 hours.

The forecasting tool uses the cxpcctcd usage profiles from the first pass algorithm to generate statistics that arc both consistent with scheduling and designed to give insight into resource and requirement problems. Many statistics can be derived, including: expected lost time and number of resources required to meet performance rcquircmcnts.

First Pass (Forecast)

In forecasting, many details arc presently ignored that would bc considered crucial to scheduling; however, the goal is to gain as much insight into scheduling problems without resorting to the lcvcl of detailed data input required for scheduling or the attendant run-time. For example, the definition of a request for forecasting is considerably simpler than that for scheduling. A forecasting request is defined by:

- View period object (e.g., spacecraft, planet, shifts)
- Usable antenna rcsourccs
- Prc- and post-calibration times
- Minimum track duration
- Average Track duration
- Number of tracks per week

The cnd result of the first pass is a set of cxpcctcd usage profiles. Onc profile is gcncratcd for each rcsourcc. Each profile rcprcscnts the cxpcctcd usage of that rcsourcc as a function of time. In addition, cxpcctcd usagc values arc subdivided by activity group as wcll. Expected usage profiles arc constructcd by looping through each group of activities that is 10 be scheduled.

The calculation of expected usage profiles is a three stage process that incorporates requirements, rcsourcc capacity, and view period information. The steps arc:

1) Calculate an *expected usage value* for each rcquircmcnt. This value represents the ratio of time that must bc schcdulcd for a requirement to the time that is available for the rcquircmcnt 10 bc scheduled.

2) The expected usage value is used with the view periods and the prc- and post-calibration timcs to generate individual expected usage profiles. Each of these profiles represents the demand from onc rcquircmcnt for each point in time for each antenna.

3) Individual expected usage profilcs for each antenna arc summcd, rcsulting in onc cxpcctcd usage profile for each antenna.

Each of these steps is dcscribcd in greater detail below.

Calculation of Exnectcd Usagc Valuc

In calculating the expected usage value for a rcquircmcnt, first the total amount of requested time is calculated. This is calculated based upon the rcquircmcnt's average duration, prc-, and post-calibration times, and the number of tracks requested, Calibration times arc included bccausc they represent demand on the rcsourccs just as the actual track.

The next step is to find all usablc view periods for the specified set of resources that are long enough to support the minimum requested duration. Then, the prc- and post-calibration times are appended to them, resulting in what is called *request slots*. Request slot time is dcfincd as the sum of all durations of these request slots. An cxpcctcd usage is then calculated and represents the total rcqucstcd time divided by total rcqucst slot time. If the cxpcctcd usage is grcatcr than one, this means that the rcquircmcnt cannot bc supported by the resource(s) spccificd. This is caused by physical constraints or resource downtime.

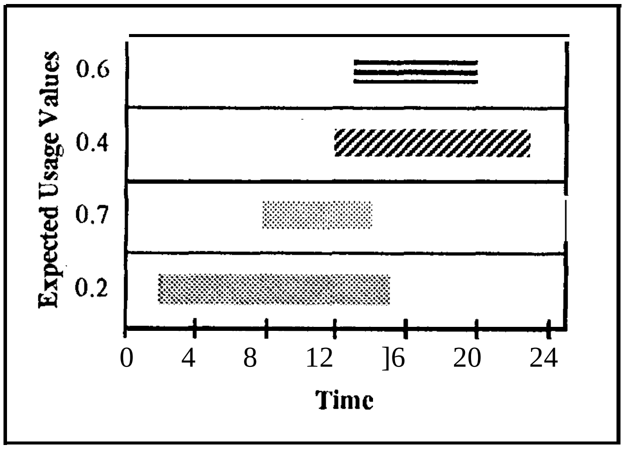

Figure 1- Example Request Slots and Usage Values

Calculation of Individual Profilcs

The next step is to generate individual cxpcctcd usage profiles. In FASTER, these profilcs are represented by step functions. The profile for a single rcqucst will bc a two level step function having the request's expected usage during all usable request slots and the value zcro at all other timcs.

Calculation of Antcnna Profilcs

The last step is to sum all individual request profiles for each antenna. This results in a complete picture of the expected usage of each antenna at all times. By using this information to drive the scheduling phase, the scheduler is able to determine what areas should bc avoided even when nothing has actually been schcdulcd in that area, A summcd profile dissection is shown in figure 2 (based upon data from figure 1). An example of a complete one week antenna profile is shown in figure 3.

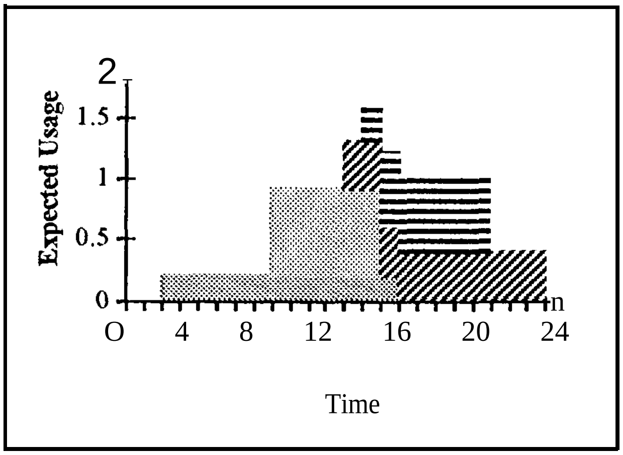

Figure 2- Example Sumation of Individual Profiles

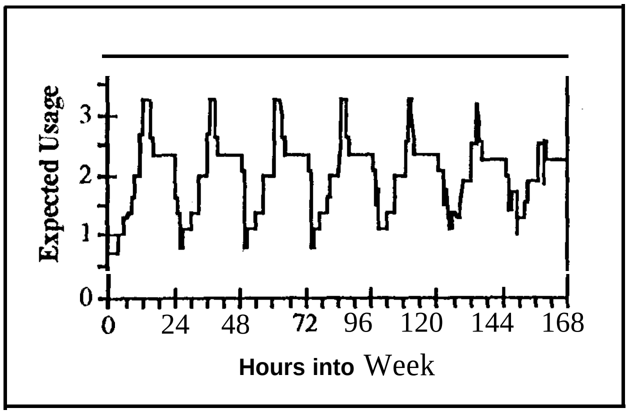

Figure 3- Example Antenna Expected Usage Profile

Sccond Pass (Schcdulc)

FASTER performs a series of optimizations steps, onc for each requirement that is scheduled. Decomposing a problcm into multiple optimization steps dots not necessarily result in a solution that is anywhere near opt imal. To mitigate the cffccts of the suboptimization, the expected usage profiles are used to derive the cost function for each optimization and thus allow the schcdulcr to "look ahead" at requests that have yet to be scheduled.

As mentioned previously, the description of a rcqucst is more detailed for scheduling. In particular, separation constraints (minimum, maximum, and preferred) are added to force a desired distribution of tracks. Separation constraints determine the time between the cnd of one track in a request and the start of the next track in the request.

The first step in scheduling a request is to remove its cent ribut ion from all expected usage profiles. Next a set of ***break points*** are calculated, indicating all points at which it would be reasonable to begin or end a track. They are called break points bccausc they correlate to breaks in thc expected usage profiles. Break points arc distinguished as to whether they are start or end break points. This approach transforms the problem from finding the best place to schedule n tracks to finding a minimum cost "path" through the correct number of break points such that all duration and separation constraints are met.

This problcm is similar to that of the classical traveling salesman problem where a shortest path must be found through a set of cities. [1] Instead of finding a path through cities, find a path through alternating start and cnd break points.

When traveling from a start break point to an end break points, the following must hold:

- the cnd point being considered must have come from the same view period as the start point
- the cnd point must bc at least thc minimum duration aftcr the start point
- the end point must bc less than the maximum duration after the start point

The cost for a valid start to end path (a track) is the combination of a duration preference value and an cxpcctcd resource cost. The expected resource cost is calculated by integrating the area under the expected usage profilc for the chosen antenna during the interval start-pre to end+post.

When traveling from an end break point to a start break point, the following must hold

- the start must bc at least the minimum separation aftcr the end
- the start must be lcss than the maximum duration after the end

The cost for a valid end to start path (gap) is calculated simply from any preferences on separation (since no rcsourccs are consumed during this time).

In addition, several other properties of the problcm can bc used to reduce the total number of paths that must bc evaluated, For example, given that all of the tracks must fit within a certain period of time (an overall window), upper and lower bounds can be calculated for the start and cnd time of each track. Another example is to calculate a simple first-cut path and use its cost to pnmc paths as they are calculated.

Once the optimal path has been calculated, the tracks of that request are actually scheduled and the expected usage profiles are updated to reflect the ncw information. Since the antcnna and start and end timcs for the tracks arc now known, a contribution of 1.0 is added in for each track. In this way, the scheduling pass can bc viewed as transforming the expected usage profiles from probabilistic to deterministic.

Limitations

While this approach to scheduling and forecasting has been successful in our particular problcm, there arc some limitations to the current approach that might limit its usefulness in other cases. These include:

- The need for "view period like" restrictions — If there arc few or no limitations on where activities may occur, profiles will be of littlc use (a flat profile is uninteresting).

- The need for durations and numbers of activities to bc known prior to scheduling (within ranges) — If the total amount of lime that might bc scheduled for a request can vary over an a order of magnitude rangc, then calculation of cxpcctcd usage values may be difficult or cxtrcmcly inaccurate,

- The need for an oversubscribed problem with requirement interaction — if requirements do not ovcrlap sufficiently, enough break points may not bc gcncratcd and feasible schedules may not be found.

Improvements

As with all solutions, there are steps that could bc taken to improve upon the approach. These include:

- Bcttcr search algorithms — little attention has been paid to reducing the number of paths that are evaluated.

- Bcttcr modeling of probability distribution — currently profiles are modeled as step functions; this is not wholly accurate. A true representation would involve piece-wise linear functions. However, it is not clear that the additional overhead would really improve schcdulc quality.

- Consider spacing constrains in the generation of cxpcctcd usage profiles — the current profile generation schcmc implies a more or lcss even distribution of activities. Depending upon view periods and spacing interaction, this may not bc a good assumption.

- Implcmcnt an incremental update feature Æ the currcnt systcm does not support incrcmcntal rcquircmcnt modifications/additions.

Implementation

The FASTER systcm runs on a 386/486 class machine with at least 8MB of RAM running Microsoft Windows version 3.1.2 FASTER requires connection to a server running Novell Netware. This server is used both for program sharing as wcll as being the database server in a client/server architecture. FASTER is implemented in C++ and based upon a set of classes named RASCL (Rcsourcc Allocation and Scheduling Class Library). RASCL was codcvclopcd by JPL and CTA (a Langley contractor). RASCL implements many of the common data structures and primitives used in scheduling and forecasting, including:

- int cgcr time rcprcscntation
- intcrvals
- timclincs
- profiles

The. forecasting portion of FASTER uses Microsoft Excel (a spreadsheet) as a tabular input sheet and for graphing statistical information. Microsoft Access is being used as a client front end to the database server for database maintenance and report generation. The use of these COTS (commercial of the shclf) packages has increased programmer productivity, allowed users to customize reports and graphs, and reduced the maintenance burden.

In terms of performance, the forecasting tool is able to generate an average one year forecast on a 33MHz 486 in under five minutes. This includes the time to retrieve approximately 30,000 view period records over the network, calculate expected usage profiles, calculate desired statistics from the profiles, and write a file with the desired statistical information for loading back into Excel.

Issues

For those who would bc interested in trying some of the approaches wc have described, in an operational environment, these issues should be addressed.

While using COTS packages has great advantages, it also can imply dependence upon that package. In particular: When a new version comes out, do you upgrade? Are there ncw capabilities that really would bc useful? Is it really backward compatible? Does the compatibility require some sort of “conversion”. Can everyone bc upgraded at once? If you don’t upgrade, can you still get technical support? As you would guess, the only way to really answer some of these questions is to try it. Install a ncw version in an off-line simulation of operations and test, tcst, test.

As our user base has grown and the dcvclopmcnt team has shrunk, training and support have bccomc prime issues. When considering a large systcm, the normal questions must be asked. Who do you train? Do you charge for training? Who do you support? Do you charge for support? Since the program makes heavy use of a particular COTS package, do you have to be the support for the COTS package, In general, this has to bc worked out with the customer and the funding source; just remember, customer support can bc a different iating factor when competing.

At JPL, as at other large companies, the demand for network scrviccs sometimes outstrips the supply.

FASTER is dependent upon the institutional network for users who arc outside the local area of the server. Network problems can become a support headache that you may be near powerless to affect. Do some testing to determine performance and bccomc involved in any planning committees that address these types of problems.

Acknowledgments

The authors wish to thank Eugene Burke of the Multimission Operations Systems Office for his continued support of FASTER development and J.R. Hall of the Office of Telecommunications and Data Acquisition for his continued support of forecasting development. U.S. Government sponsorship under eont ract NAS7-918 is acknowledged.

References

[1] Cormen, Thomas H., Charles E. Leiserson and Ronald L. Rivest,*Introduction to Algorithms,* MIT Press, 1990.

[2] Werntz, David, Silvino Zendejas and Steven Loyola, *Forecasting And Scheduling Tool for Earth-based Resources (FASTER) User Guide, 1992,* JPL Internal Document D-10450.